

1 What is claimed is:

2  
3 1. A method of operating a snapshot copy facility that stores a plurality of  
4 snapshot copies of a production file system, each of the snapshot copies being a prior  
5 state of the production file system at a respective point in time, said method comprising:

6 the snapshot copy facility receiving a request for the difference between a  
7 specified older one of the snapshot copies and a specified younger one of the snapshot  
8 copies; and

9 the snapshot copy facility responding to the request by returning the difference  
10 between the specified older one of the snapshot copies and the specified younger one of  
11 the snapshot copies.

12  
13 2. The method as claimed in claim 1, wherein the production file system  
14 includes blocks of data, and the snapshot copy facility returns an identification of each  
15 block that has changed between the specified older one of the snapshot copies and the  
16 specified younger one of the snapshot copies, and the snapshot copy facility returns the  
17 data in the specified younger one of the snapshot copies for said each block that has  
18 changed between the specified older one of the snapshot copies and the specified younger  
19 one of the snapshot copies.

20  
21 3. The method as claimed in claim 2, wherein the identifications of the  
22 changed blocks and the data of the changed blocks are returned in a sequential block  
23 number order.

1

2           4.     The method as claimed in claim 1, wherein the snapshot copy facility has  
3     an index for each snapshot copy for indicating changes between said each snapshot copy  
4     and a next snapshot copy of the production file system, and the method includes scanning  
5     the index for the specified older one of the snapshot copies.

6

7           5.     The method as claimed in claim 4, wherein the index for at least one of the  
8     snapshot copies is a bit map.

9

10          6.     The method as claimed in claim 4, wherein the index for at least one of the  
11     snapshot copies includes a hash table.

12

13          7.     The method as claimed in claim 4, which includes scanning the indices for  
14     a sequence of the snapshot copies including the index for the specified older one of the  
15     snapshot copies and a respective index for each snapshot copy of the production file  
16     system that is both younger than the specified older one snapshot copies and older than  
17     the specified younger one of the snapshot copies.

18

19          8.     The method as claimed in claim 7, wherein the indices for the sequence of  
20     the snapshot copies are scanned by a program routine having an outer loop indexing  
21     blocks of data in the file system, and an inner loop indexing the snapshot copies in the  
22     sequence of the snapshot copies.

23

1           9.     The method as claimed in claim 1, wherein the snapshot copy facility has  
2     an index for each snapshot copy for indicating blocks of data that are known to be invalid  
3     in said each snapshot copy, and the method includes scanning the index for the specified  
4     younger one of the snapshot copies, and when the index indicates that a block is not  
5     known to be invalid, then determining whether the block has changed between the  
6     specified older one of the snapshot copies and the specified younger one of the snapshot  
7     copies.

8  
9           10.    A method of operating a snapshot copy facility that stores a plurality of  
10    snapshot copies of a production file system, each of the snapshot copies being a prior  
11    state of the production file system at a respective point in time, the snapshot copy facility  
12    having an index for each snapshot copy for indicating blocks of data in the production  
13    file system that have changed between said each snapshot copy and a next snapshot copy  
14    of the production file system, wherein the method comprises:

15                scanning the indices for a sequence of the snapshot copies to determine the  
16    blocks that have changed between an older one of the snapshot copies and a younger one  
17    of the snapshot copies, the sequence of the snapshot copies including the older one of the  
18    snapshot copies and each of the snapshot copies that is both younger than the older one of  
19    the snapshot copies and older than the younger one of the snapshot copies.

20  
21           11.    The method as claimed in claim 10, wherein at least one of the indices is a  
22    bit map.

1           12.    The method as claimed in claim 10, wherein at least one of the indices  
2 includes a hash table.

3  
4           13.    The method as claimed in claim 10, which includes responding to a  
5 request for the difference between the older one of the snapshot copies and a younger one  
6 of the snapshot copies by:

7                returning a sequence of block numbers of the blocks that have changed between  
8 the older one of the snapshot copies and the younger one of the snapshot copies, and

9                returning the data in the younger one of the snapshot copies for the blocks that  
10 have changed between the older one of the snapshot copies and the younger one of the  
11 snapshot copies.

12  
13           14.    The method as claimed in claim 13, wherein the block numbers of the  
14 changed blocks and the data of the changed blocks are returned in a sequential block  
15 number order.

16  
17           15.    The method as claimed in claim 10, wherein the indices for the sequence  
18 of the snapshot copies are scanned by a program routine having an outer loop indexing  
19 respective blocks, and an inner loop indexing snapshot copies in the sequence of the  
20 snapshot copies.

21  
22           16.    The method as claimed in claim 15, wherein the snapshot copy facility has  
23 a meta bit map for each snapshot copy for indicating blocks of data that are known to be

1 invalid in said each snapshot copy, and the method includes scanning the meta bit map  
2 for the specified younger one of the snapshot copies, and when the meta bit map is found  
3 to indicate that a block is not known to be invalid, then determining whether the block  
4 has changed between the specified older one of the snapshot copies and the specified  
5 younger one of the snapshot copies by scanning the indices for the sequence of the  
6 snapshot copies.

7  
8 17. A method of operating a snapshot copy facility that stores a plurality of  
9 snapshot copies of a production file system, each of the snapshot copies being a prior  
10 state of the production file system at a respective point in time, the snapshot copy facility  
11 having a first index for each snapshot copy for indicating blocks of data in the production  
12 file system that have changed between said each snapshot copy and a next snapshot copy  
13 of the production file system and that have a "before image" saved for said each snapshot  
14 copy, the snapshot copy facility having a second index for said each snapshot copy for  
15 indicating blocks of data that are not in use in said each snapshot copy; said method  
16 comprising:

17 responding to a request for the difference between a specified older one of the  
18 snapshot copies and a specified younger one of the snapshot copies by accessing the  
19 second index for the specified younger one of the snapshot copies to determine blocks of  
20 data in the production file system that are in use in the specified younger one of the  
21 snapshot copies, and for blocks of data in the production file system that are in use in the  
22 specified younger one of the snapshot copies, accessing at least one of the first indices for  
23 a sequence of the snapshot copies to determine blocks that have changed between an

1 older one of the snapshot copies and a younger one of the snapshot copies, the sequence  
2 of the snapshot copies including the older one of the snapshot copies and each of the  
3 snapshot copies that is both younger than the older one of the snapshot copies and older  
4 than the younger one of the snapshot copies.

5  
6 18. The method as claimed in claim 17, which also includes accessing at least  
7 one of the second indices for the snapshot copies in the sequence of the snapshot copies  
8 and finding that at least one of the blocks is not in use in at least one of the snapshot  
9 copies in the sequence of the snapshot copies to determine that said at least one of the  
10 blocks has changed between the older one of the snapshot copies and the younger one of  
11 the snapshot copies not changed.

12  
13 19. A method of operating a network file server, the network file server  
14 having a snapshot copy facility for storing a plurality of snapshot copies of a production  
15 file system, each of the snapshot copies being a prior state of the production file system at  
16 a respective point in time, said method comprising:

17 the network file server receiving a request for an update to a specified snapshot  
18 copy of the production file system;

19 the network file server responding to the request by checking whether the  
20 snapshot copy facility contains the specified snapshot copy of the production file system,  
21 and upon finding that the snapshot copy facility contains the specified snapshot copy of  
22 the production file system, the network file server returning the difference between the

1 specified snapshot copy of the production file system and a more recent snapshot copy of  
2 the production file system.

3  
4 20. The network file server as claimed in claim 19, wherein the more recent  
5 snapshot copy of the production file system is the most recent one of the snapshot copies  
6 of the production file system that are stored in the snapshot copy facility.

7  
8 21. The network file server as claimed in claim 19, wherein the request  
9 specifies the more recent snapshot copy of the production file system.

10  
11 22. The network file server as claimed in claim 19, wherein the network file  
12 server returns the difference between the specified snapshot copy of the production file  
13 system and the more recent snapshot copy of the production file system by returning a  
14 series of block numbers for blocks of the production file system that have changed  
15 between the specified snapshot copy of the production file system and the more recent  
16 snapshot copy of the production file system, and returning the data in the more recent  
17 snapshot copy of the production file system for said each block that has changed between  
18 the specified one of the snapshot copies of the production file system and the more recent  
19 snapshot copy of the production file system.

20  
21 23. In a data processing network having a client and a network file server, the  
22 network file server storing a plurality of snapshot copies of a production file system, each  
23 of the snapshot copies being a prior state of the production file system at a respective

1 point in time, the client having a local version of an older one of the snapshot copies, a  
2 method of providing the client with a younger one of the snapshot copies, the method  
3 comprising:

4 the network file server determining the difference between the younger one of the  
5 snapshot copies and the older one of the snapshot copies;

6 the network file server transmitting the difference between the younger one of the  
7 snapshot copies and the older one of the snapshot copies to the local version of the older  
8 one of the snapshot copies; and

9 writing the difference between the younger one of the snapshot copies and the  
10 older one of the snapshot copies into the local version of the older one of the snapshot  
11 copies to produce a local version of the younger one of the snapshot copies.

12  
13 24. The method as claimed in claim 23, wherein the network file server  
14 determines the difference between the younger one of the snapshot copies and the older  
15 one of the snapshot copies in response to an update request from the client, the update  
16 request specifying the older one of the snapshot copies.

17  
18 25. The method as claimed in claim 23, wherein the network file server  
19 determines the difference between the younger one of the snapshot copies and the older  
20 one of the snapshot copies by determining blocks of the production file system that have  
21 changed between the younger one of the snapshot copies and the older one of the  
22 snapshot copies, and the network file server transmits the difference between the younger  
23 one of the snapshot copies and the older one of the snapshot copies to the local version of



1 the younger one of the snapshot copies by transmitting block identifiers for the blocks of  
2 the production file system that have changed between the younger one of the snapshot  
3 copies and the older one of the snapshot copies, and by transmitting the data in the  
4 younger one of the snapshot copies for the blocks of the production file system that have  
5 changed between the younger one of the snapshot copies and the older one of the  
6 snapshot copies.

7  
8 26. A snapshot copy facility comprising:

9 storage for storing a plurality of snapshot copies of a production file system, each  
10 of the snapshot copies being a prior state of the production file system at a respective  
11 point in time; and

12 at least one processor programmed for receiving a request for the difference  
13 between a specified older one of the snapshot copies and a specified younger one of the  
14 snapshot copies; and for responding to the request by returning the difference between  
15 the specified older one of the snapshot copies and the specified younger one of the  
16 snapshot copies.

17  
18 27. The snapshot copy facility as claimed in claim 26, wherein the production  
19 file system includes blocks of data, and said at least one processor is programmed for  
20 returning an identification of each block that has changed between the specified older one  
21 of the snapshot copies and the specified younger one of the snapshot copies, and the  
22 snapshot copy facility returns the data in the specified younger one of the snapshot copies

1 for said each block that has changed between the specified older one of the snapshot  
2 copies and the specified younger one of the snapshot copies.

3

4 28. The snapshot copy facility as claimed in claim 27, wherein said at least  
5 one processor is programmed to return the identifications of the changed blocks and the  
6 data of the changed blocks in a sequential block number order.

7

8 29. The snapshot copy facility as claimed in claim 26, wherein the snapshot  
9 copy facility has an index for each snapshot copy for indicating changes between said  
10 each snapshot copy and a next snapshot copy of the production file system, and said at  
11 least one processor is programmed for scanning the index for the specified older one of  
12 the snapshot copies.

13

14 30. The snapshot copy facility as claimed in claim 29, wherein the index for at  
15 least one of the snapshot copies is a bit map.

16

17 31. The snapshot copy facility as claimed in claim 29, wherein the index for at  
18 least one of the snapshot copies includes a hash table.

19

20 32. The snapshot copy facility as claimed in claim 29, wherein said at least  
21 one processor is programmed for scanning the indices for a sequence of the snapshot  
22 copies including the index for the specified older one of the snapshot copies and a  
23 respective index for each snapshot copy of the production file system that is both younger

1 than the specified older one snapshot copies and older than the specified younger one of  
2 the snapshot copies.

3  
4 33. The snapshot copy facility as claimed in claim 32, wherein said at least  
5 one processor is programmed for scanning the indices for the sequence of the snapshot  
6 copies by a program routine having an outer loop indexing the blocks, and an inner loop  
7 indexing the snapshot copies in the sequence of the snapshot copies.

8  
9 34. The snapshot copy facility as claimed in claim 26, wherein the snapshot  
10 copy facility has an index for each snapshot copy for indicating blocks of data that are  
11 known to be invalid in said each snapshot copy, and said at least one processor is  
12 programmed for scanning the index for the specified younger one of the snapshot copies,  
13 and when the index indicates that a block is not known to be invalid, then determining  
14 whether the block has changed between the specified older one of the snapshot copies  
15 and the specified younger one of the snapshot copies.

16  
17 35. A snapshot copy facility comprising:  
18 storage for storing a plurality of snapshot copies of a production file system, each  
19 of the snapshot copies being a prior state of the production file system at a respective  
20 point in time;

21 an index for each snapshot copy for indicating blocks of data in the production  
22 file system that have changed between said each snapshot copy and a next snapshot copy  
23 of the production file system, and

1           at least one processor programmed for scanning the indices for a sequence of the  
2       snapshot copies to determine the blocks that have changed between an older one of the  
3       snapshot copies and a younger one of the snapshot copies, the sequence of the snapshot  
4       copies including the older one of the snapshot copies and each of the snapshot copies that  
5       is both younger than the older one of the snapshot copies and older than the younger one  
6       of the snapshot copies.

7  
8           36.     The snapshot copy facility as claimed in claim 35, wherein at least one of  
9       the indices is a bit map.

10  
11          37.     The snapshot copy facility as claimed in claim 35, wherein at least one of  
12       the indices includes a hash table.

13  
14          38.     The snapshot copy facility as claimed in claim 35, wherein the production  
15       file system includes blocks of data, and said at least one processor is programmed to  
16       respond to a request for the difference between the older one of the snapshot copies and a  
17       younger one of the snapshot copies by:

18               returning a sequence of block numbers of the blocks that have changed between  
19       the older one of the snapshot copies and the younger one of the snapshot copies, and

20               returning the data in the younger one of the snapshot copies for the blocks that  
21       have changed between the older one of the snapshot copies and the younger one of the  
22       snapshot copies.

1           39.     The snapshot copy facility as claimed in claim 38, wherein said at least  
2     one processor is programmed to return the block numbers of the changed blocks and the  
3     data of the changed blocks in a sequential block number order.

4  
5           40.     The snapshot copy facility as claimed in claim 35, wherein said at least  
6     one processor is programmed for scanning the indices for the sequence of the snapshot  
7     copies by a program routine having an outer loop indexing the blocks, and an inner loop  
8     indexing the snapshot copies in the sequence of the snapshot copies.

9  
10          41.     The snapshot copy facility as claimed in claim 35, which includes a meta  
11     bit map for each snapshot copy for indicating blocks of data that are known to be invalid  
12     in said each snapshot copy, and wherein said at least one processor is programmed for  
13     scanning the meta bit map for the specified younger one of the snapshot copies, and when  
14     the meta bit map is found to indicate that a block is not known to be invalid, then  
15     determining whether the block has changed between the specified older one of the  
16     snapshot copies and the specified younger one of the snapshot copies by scanning the  
17     indices for the sequence of the snapshot copies.

18  
19  
20          42.     A snapshot copy facility comprising:  
21             storage for storing a plurality of snapshot copies of a production file system, each  
22     of the snapshot copies being a prior state of the production file system at a respective  
23     point in time;

1 a first index for each snapshot copy for indicating blocks of data in the production  
2 file system that have changed between said each snapshot copy and a next snapshot copy  
3 of the production file system and that have a "before image" for said each snapshot copy  
4 stored in the storage,

5 a second index for each snapshot copy for indicating blocks of data that are not in  
6 use in said each snapshot copy, and

7 at least one processor programmed for responding to a request for the difference  
8 between a specified older one of the snapshot copies and a specified younger one of the  
9 snapshot copies by accessing the second index for the specified younger one of the  
10 snapshot copies to determine blocks of data in the production file system that are in use in  
11 the specified younger one of the snapshot copies, and for blocks of data in the production  
12 file system that are in use in the specified younger one of the snapshot copies, accessing  
13 at least one of the first indices for a sequence of the snapshot copies to determine blocks  
14 that have changed between an older one of the snapshot copies and a younger one of the  
15 snapshot copies, the sequence of the snapshot copies including the older one of the  
16 snapshot copies and each of the snapshot copies that is both younger than the older one of  
17 the snapshot copies and older than the younger one of the snapshot copies.

18  
19 43. The snapshot copy facility as claimed in claim 42, wherein said at least  
20 one processor is also programmed for accessing at least one of the second indices for the  
21 snapshot copies in the sequence of the snapshot copies and finding that at least one of the  
22 blocks is not in use in at least one of the snapshot copies in the sequence of the snapshot

1 copies to determine that said at least one of the blocks has changed between the older one  
2 of the snapshot copies and the younger one of the snapshot copies not changed.

3  
4 44. A network file server comprising a snapshot copy facility for storing a  
5 plurality of snapshot copies of a production file system, each of the snapshot copies being  
6 a prior state of the production file system at a respective point in time,

7 wherein the network file server is programmed for receiving a request for an  
8 update to a specified snapshot copy of the production file system, and responding to the  
9 request by checking whether the snapshot copy facility contains the specified snapshot  
10 copy of the production file system, and upon finding that the snapshot copy facility  
11 contains the specified snapshot copy of the production file system, returning the  
12 difference between the specified snapshot copy of the production file system and a more  
13 recent snapshot copy of the production file system.

14  
15 45. The network file server as claimed in claim 44, wherein the more recent  
16 snapshot copy of the production file system is the most recent one of the snapshot copies  
17 of the production file system that are stored in the snapshot copy facility.

18  
19 46. The network file server as claimed in claim 44, wherein the request  
20 specifies the more recent snapshot copy of the production file system.

21  
22 47. The network file server as claimed in claim 44, wherein the network file  
23 server is programmed to return the difference between the specified snapshot copy of the

1 production file system and the more recent snapshot copy of the production file system  
2 by returning a series of block numbers for blocks of the production file system that have  
3 changed between the specified snapshot copy of the production file system and the more  
4 recent snapshot copy of the production file system, and the data in the more recent  
5 snapshot copy of the production file system for said each block that has changed between  
6 the specified one of the snapshot copies of the production file system and the more recent  
7 snapshot copy of the production file system.

8  
9 48. The network file server as claimed in claim 44, wherein the network file  
10 server is programmed to return the more recent snapshot copy of the production file  
11 system upon finding that the snapshot copy facility does not contain the specified  
12 snapshot copy of the production file system.

13  
14 49. A program storage device containing a program for a snapshot copy  
15 facility, the snapshot copy facility storing a plurality of snapshot copies of a production  
16 file system, each of the snapshot copies being a prior state of the production file system at  
17 a respective point in time, wherein the program is executable for responding to a request  
18 for the difference between a specified older one of the snapshot copies and a specified  
19 younger one of the snapshot copies by returning the difference between the specified  
20 older one of the snapshot copies and the specified younger one of the snapshot copies.

21  
22 50. The program storage device as claimed in claim 49, wherein the program  
23 is executable for returning an identification of each block that has changed between the



1 specified older one of the snapshot copies and the specified younger one of the snapshot  
2 copies, and for returning the data in the specified younger one of the snapshot copies for  
3 said each block that has changed between the specified older one of the snapshot copies  
4 and the specified younger one of the snapshot copies.

5  
6 51. The program storage device as claimed in claim 50, wherein the program  
7 is executable for returning the identifications of the changed blocks and the data of the  
8 changed blocks in a sequential block number order.

9  
10 52. The program storage device as claimed in claim 49, wherein the snapshot  
11 copy facility has an index for each snapshot copy for indicating changes between said  
12 each snapshot copy and a next snapshot copy of the production file system, and the  
13 program is executable for scanning the index for the specified older one of the snapshot  
14 copies.

15  
16 53. The program storage device as claimed in claim 52, wherein the program  
17 is executable for scanning the indices for a sequence of the snapshot copies including the  
18 index for the specified older one of the snapshot copies and a respective index for each  
19 snapshot copy of the production file system that is both younger than the specified older  
20 one snapshot copies and older than the specified younger one of the snapshot copies.

21  
22 54. The program storage device as claimed in claim 53, wherein the program  
23 is executable for scanning the indices for the sequence of the snapshot copies by a

1 program routine having an outer loop indexing the blocks, and an inner loop indexing the  
2 snapshot copies in the sequence of the snapshot copies.

3  
4 55. A program storage device containing a program for a snapshot copy  
5 facility, the snapshot copy facility having a plurality of snapshot copies of a production  
6 file system, each of the snapshot copies being a prior state of the production file system at  
7 a respective point in time, and an index for each snapshot copy for indicating blocks of  
8 data in the production file system that have changed between said each snapshot copy and  
9 a next snapshot copy of the production file system, wherein the program is executable for  
10 scanning the indices for a sequence of the snapshot copies to determine the blocks that  
11 have changed between an older one of the snapshot copies and a younger one of the  
12 snapshot copies, the sequence of the snapshot copies including the older one of the  
13 snapshot copies and each of the snapshot copies that is both younger than the older one of  
14 the snapshot copies and older than the younger one of the snapshot copies.

15  
16 56. The program storage device as claimed in claim 55, wherein the program  
17 is executable for responding to a request for the difference between the older one of the  
18 snapshot copies and a younger one of the snapshot copies by:

19 returning a sequence of block numbers of the blocks that have changed between  
20 the older one of the snapshot copies and the younger one of the snapshot copies, and

21 returning the data in the younger one of the snapshot copies for the blocks that  
22 have changed between the older one of the snapshot copies and the younger one of the  
23 snapshot copies.

1  
2 57. The program storage device as claimed in claim 56, wherein the program  
3 is executable for returning the block numbers of the changed blocks and the data of the  
4 changed blocks in a sequential block number order.  
5

6 58. The program storage device as claimed in claim 55, wherein the program  
7 is executable for scanning the indices for the sequence of the snapshot copies by a  
8 program routine having an outer loop indexing the blocks, and an inner loop indexing the  
9 snapshot copies in the sequence of the snapshot copies.  
10

11 59. The program storage device as claimed in claim 55, wherein the snapshot  
12 copy facility has a meta bit map for each snapshot copy for indicating blocks of data that  
13 are known to be invalid in said each snapshot copy, and wherein the program storage  
14 device is executable for scanning the meta bit map for the specified younger one of the  
15 snapshot copies, and when the meta bit map is found to indicate that a block is not known  
16 to be invalid, then determining whether the block has changed between the specified  
17 older one of the snapshot copies and the specified younger one of the snapshot copies by  
18 scanning the indices for the sequence of the snapshot copies.  
19

20 60. A program storage device containing a program for a snapshot copy  
21 facility, the snapshot copy facility having a plurality of snapshot copies of a production  
22 file system, each of the snapshot copies being a prior state of the production file system at  
23 a respective point in time, a first index for each snapshot copy for indicating blocks of

1 data in the production file system that have changed between said each snapshot copy and  
2 a next snapshot copy of the production file system and that have a "before image" for  
3 said each snapshot copy stored in the snapshot copy facility, and a second index for each  
4 snapshot copy for indicating blocks of data that are not in use in said each snapshot copy,  
5 wherein the program is executable for responding to a request for the difference between  
6 a specified older one of the snapshot copies and a specified younger one of the snapshot  
7 copies by accessing the second index for the specified younger one of the snapshot copies  
8 to determine blocks of data in the production file system that are in use in the specified  
9 younger one of the snapshot copies, and for blocks of data in the production file system  
10 that are in use in the specified younger one of the snapshot copies, accessing at least one  
11 of the first indices for a sequence of the snapshot copies to determine blocks that have  
12 changed between an older one of the snapshot copies and a younger one of the snapshot  
13 copies, the sequence of the snapshot copies including the older one of the snapshot copies  
14 and each of the snapshot copies that is both younger than the older one of the snapshot  
15 copies and older than the younger one of the snapshot copies.

16  
17 61. The program storage device as claimed in claim 60, wherein the program  
18 is executable for accessing at least one of the second indices for the snapshot copies in  
19 the sequence of the snapshot copies and finding that at least one of the blocks is not in  
20 use in at least one of the snapshot copies in the sequence of the snapshot copies to  
21 determine that said at least one of the blocks has changed between the older one of the  
22 snapshot copies and the younger one of the snapshot copies not changed.

23

1  
2           62.     A program storage device containing a program for a network file server,  
3     the network file server including a snapshot copy facility for storing a plurality of  
4     snapshot copies of a production file system, each of the snapshot copies being a prior  
5     state of the production file system at a respective point in time,

6           wherein the program is executable for receiving a request for an update to a  
7     specified snapshot copy of the production file system, and responding to the request by  
8     checking whether the snapshot copy facility contains the specified snapshot copy of the  
9     production file system, and upon finding that the snapshot copy facility contains the  
10    specified snapshot copy of the production file system, returning the difference between  
11    the specified snapshot copy of the production file system and a more recent snapshot  
12    copy of the production file system.

13  
14           63.     The program storage device as claimed in claim 62, wherein the more  
15    recent snapshot copy of the production file system is the most recent one of the snapshot  
16    copies of the production file system that are stored in the snapshot copy facility.

17  
18           64.     The program storage device as claimed in claim 62, wherein the request  
19    specifies the more recent snapshot copy of the production file system.

20  
21           65.     The program storage device as claimed in claim 62, wherein the program  
22    is executable for returning the difference between the specified snapshot copy of the  
23    production file system and the more recent snapshot copy of the production file system

1 by returning a series of block numbers for blocks of the production file system that have  
2 changed between the specified snapshot copy of the production file system and the more  
3 recent snapshot copy of the production file system, and returning the data in the more  
4 recent snapshot copy of the production file system for said each block that has changed  
5 between the specified one of the snapshot copies of the production file system and the  
6 more recent snapshot copy of the production file system.

7  
8 66. The program storage device as claimed in claim 62, wherein the program  
9 is executable for returning the more recent snapshot copy of the production file system  
10 upon finding that the snapshot copy facility does not contain the specified snapshot copy  
11 of the production file system.

12